Python in Finance: Case Studies for Today's Changing Landscape

Supplementary 3rd Semester Study Abroad Project

at

University of Applied Sciences Lucerne

3rd Semester, M.Sc. Banking and Finance

Submitted by:

Jean-Philipp Brühwiler

Lucerne, December 2023

University of Applied Sciences Lucerne

Master of Science in Banking and Finance

Supplementary 3rd Semester Study Abroad Project

December 2023

Author:

Jean-Philipp Brühwiler, Dorfplatz 3 8810 Horgen, jean-philipp.bruehwiler@stud.hslu.ch

Supervisors:

Mr. Moreno Frigg, Suurstoffi 1 6343 Rotkreuz, moreno.frigg@hslu.ch

Dr. Simon Amrein, Suurstoffi 1 6343 Rotkreuz, simon.amrein@hslu.ch

Table of Contents

1. Abstract

The purpose of this research paper is to examine the overall technological trends influencing finance and the skills needed for aspiring professionals to remain competitive in a rapidly changing environment. This paper demonstrates that Python, a general-purpose scripting language, is the key technical skill students should learn to stay ahead. The reasons for this are its widespread industry adoption, is unlikely to be replaced in the future, can be learned in a reasonable amount of time, and integrates well into existing enterprise infrastructure. This paper reviews the research that shows how building projects is one of the best ways to lean Python. Then, the paper outlines three possible case study project with industry relevant applications for finance students. The case studies include a binary classification case to predict credit default, simulating portfolio returns and volatility with a Monte Carlo simulation, and a cross-sectional return analysis case. Finally, the paper explores which existing HSLU Msc. Banking and Finance modules the cases can be incorporated into and possible limitations of the paper's findings.

2. Introduction

This section provides the reader with a summary of contemporary technological trends that are influencing the financial industry, placing particular emphasis on how these trends impact individuals employed within the financial sector. The analysis then explores the skills financial practitioners need to learn to remain competitive in an increasingly digital landscape. Finally, the section ends by showing that Python is the key technical skill financial practitioners should learn within the outlined context.

2.1 Change is Here

Ray Dalio is an American hedge fund manager and founder of the investment management firm Bridgewater Associates based out of Stamford, CT.[1] As of December 2023, Mr. Dalio's core hedge fund is still the largest hedge fund in the world with about $100 billion AUM (Assets Under Management).[2] Despite Mr. Dalio's undeniable success at raising capital, he is possibly better known for his book *Principles*, published in 2020.[3] The book is part autobiography and part self-help guide giving readers tips and mantras to structure their lives and finances around. Much like Mr. Dalio's hedge funds, the book has been a huge success selling over two million copies.[4]

However, with attention comes scrutiny, and in November of 2023, a new book about Mr. Dalio was published called *The Fund* by American author Rob Copeland.[5] It is essentially an unauthorized biography and critique of Mr. Dalio's principles and his controversial management style, which includes near constant brutally honest performance reviews.[6] To this end, Mr.

[1] Bridgewater Associates, "Ray Dalio — Bridgewater Associates," Bridgewater (Bridgewater Associates), accessed December 17, 2023, https://www.bridgewater.com/people/ray-dalio.
[2] Christine Williamson, "Bridgewater Retains Top Spot as Largest Hedge Fund, despite AUM Sliding 23%," Pension&Investments (Pension&Investments, September 25, 2023), https://www.pionline.com/special-report-hedge-funds/bridgewater-retains-top-spot-largest-hedge-fund-despite-aum-sliding-23.
[3] Noteworthy Nonsense, "Book Summary: Principles by Ray Dalio," Noteworthy Nonsense (Noteworthy Nonsense, September 18, 2020), https://noteworthynonsense.com/blog/09/2020/ray-dalio-principles-book-summary.
[4] Brit Farmer, "Ray Dalio Explains His Principles - 60 Minutes - CBS News," www.cbsnews.com, April 7, 2019, https://www.cbsnews.com/news/ray-dalio-explains-his-principles-60-minutes/.
[5] Rob Copeland, *The Fund: Ray Dalio, Bridgewater Associates, and the Unraveling of a Wall Street Legend*, *Google Books*(St. Martin's Publishing Group, 2023), https://books.google.com/books?id=MM2oEAAAQBAJ&source=gbs_book_other_versions.
[6] Kathleen Elkins, "Why One Billionaire Founder Will Fire Employees for Not Criticizing Him," CNBC, February 1, 2017, https://www.cnbc.com/2017/02/01/why-one-billionaire-founder-will-fire-employees-for-not-criticizing-him.html.

Copeland argues that not only are Mr. Dalio's principles not effective, but that his hedge fund is not nearly as successful as many outside observers believe it to be. Mr. Copeland's skepticism is warranted, the core fund at Bridgewater Associates has underperformed the S&P 500 for several years and has seen heavy outflows, especially in the last two years.[7]

In a recent interview about his book, Mr. Copeland discussed a conversation he had with a top Wall Street trader about Bridgewater Associates and Ray Dalio. Apparently, this trader went to Bridgewater for an interview about joining the firm and was surprised to see how "low-tech" the operation was. According to this trader, he was shocked by seeing huge stacks of paper strewn about the office and all financial analysis still being done in Excel.[8] The trader was indignant at the notion that the largest hedge fund in the world would *still* be using Excel. He described it as shocking to see this highly respected titan of Wall Street and supposed market savant still using such "outdated" tools since Wall Street's top funds had moved on from using Excel. For someone who preaches being "radically open minded" as one of his core principle to live life by, Mr. Dalio does not seem to apply this principle when it comes to updating Bridgewater's IT systems. [10]

To argue that Bridgewater is underperforming in the market because they are using Excel instead of some other technology is of course foolish. However, the fact that the largest hedge fund in the world is still using paper and Excel highlights a common problem in finance. For a long time, it has been a poorly kept secret that many financial institutions have outdated IT systems and have been very slow to update their systems and processes.[11]

There are many reasons for the finance industry's lackluster levels of digitization. For one, the sensitive nature of banking data makes any large scale update a significant security risk. A bank is built on trust, and it only takes one mistake of mishandling customer data to lose it. Additionally, there is often a lack of in-house talent with the skills to implement digital solutions. Technical skills such as cloud computing, programming, and IT are not taught in many finance programs today. This is all to say that Bridgewater is not the exception to the rule, but indicative of a norm. Financial institutions that are not adopting new technologies are falling behind – quickly – while those that do are reaping the rewards.[12]

When examining which industries are being most affected by technological change, finance is perhaps uniquely poised to face pressure.[13] The reasons are clear. Finance is data-driven industry with many repetitive tasks which are logical candidates for automation. For example, many aspects of wealth management, compliance, customer service, and investment analysis all involve many highly repetitive tasks. Additionally, more complex business structures and regulatory pressure are forcing banks to adopt technological solutions to remain efficient.[15] With a seemingly ever

---

[7] Samuel Renotte and Luciano Viterale, "Ray Dalio Portfolio - Bridgewater Associates Latest Holdings," https://tickernerd.com/, November 29, 2023, https://www.tickernerd.com/hedge-funds/ray-dalio-portfolio-bridgewater-associates/.
[8] Patrick Boyle, "Are Ray Dalio's Principles the Secret to His Success?," www.youtube.com, November 7, 2023, https://www.youtube.com/watch?v=bb5kngSrrsw.
[10] Nat Eliason, "Principles by Ray Dalio: Summary, Notes, and Lessons," Nat Eliason, 2022, https://www.nateliason.com/notes/principles-ray-dalio#:~:text=Don%27t%20confuse%20what%20you.
[11] Bryan Yurcan, "Banks' Digital Transformation Pace Is Too Slow for Consumers," The Financial Brand, August 8, 2022, https://thefinancialbrand.com/news/digital-transformation-banking/banks-digital-transformation-pace-is-too-slow-for-consumers-150467/.
[12] Oddmund Groette, "How Jim Simons' Trading Strategies Made 66% a Year (Medallion Fund Algorithm), Quantified Strategies," https://www.quantifiedstrategies.com/, December 11, 2023, https://www.quantifiedstrategies.com/jim-simons/.
[13] Diana Drake, "3 Ways Technology Is Revolutionizing Financial Services," Wharton Global Youth Program, March 3, 2021, https://globalyouth.wharton.upenn.edu/articles/business/3-ways-technology-revolutionizing-financial-services/.
[15] Christopher Woolard, Marc Saidenberg, and Eugène Goyne, "EY Global Financial Services Regulatory Outlook 2023," www.ey.com, January 18, 2023, https://www.ey.com/en_us/financial-services/regulatory-outlook.

increasing cost of compliance, banks needs to digitize in order to properly manage the regulatory requirements.

So, is it over for humans? Are we all doomed to be replaced by math equations and a cloud server? No, but to pretend nothing will change is also foolish. There is a notorious quote by American author Stewart Brand that captures this notion: *"once a new technology rolls over you, if you're not part of the steamroller, you're part of the road."* Banks, and by extension, those that work in them, find themselves in an interesting moment. It is misleading to say financial professionals will be replaced and become "part of the road," rather, the skills needed to stay competitive are changing - *quickly*.

In an episode of the *KPMG Expert Talk* podcast, the hosts spoke with Grégoire Bordier, the Chairman of the Association of Swiss Private Banks.[16] Throughout the episode, Mr. Bordier made several interesting points. Firstly, he discussed how many small-to-mid sized banks are being consolidated because they individually can't afford to compete technologically with larger institutions. Additionally, many banks don't have the in-house talent or knowledge to build digital solutions like mobile apps or robust online banking interfaces, meaning their development and maintenance must be outsourced. This can be an effective strategy, but often comes at the cost of customization and efficiency.

Many of those working in the aforementioned smaller banks will become redundant and inevitably lose their jobs. Even large banks are not immune either. After the Credit Suisse and UBS merger, it is projected that thousands of jobs will be cut.[17] Both UBS and Credit Suisse have robust online banking and mobile applications, but the layoffs highlight the often-substantial headcount reductions that follow a merger. Therefore, it is not only the direct consequences of automation that finance professionals need to be aware of, but the further consolidation of the industry which is itself precipitated by technological change. This feedback loop of digitization and consolidation means finance professionals need more arrows in their quivers to remain competitive.

Overall, Mr. Bordier's interview was very insightful, and he is clearly an expert in his field, but he did make one observation that really missed the mark. To paraphrase, towards the middle of the interview, he asserted that there was a limit to digitization in banking and there is only "so much" that can be digitized or automated. He is not wrong, there is clearly some theoretical limit to what can be automated and digitized. However, from his tone and following comments, it gave the impression he felt the industry was close to reaching this point. The evidence seems to differ from this sentiment and many banks still have a long way to go.[18]

To inject a personal aside, the last time I set foot in a bank branch or spoke with someone who worked at a bank was because I had no other choice and had to sign some documents in person. That was close to two years ago at the time I am writing. Since then, every interaction I have had with a financial institution has been completely digital and automated. Of course, my experience is strictly anecdotal and shouldn't be taken as anything more than that. However, it is undeniable

[16] KPMG Expert Talk, "A Vintage Year for Swiss Private Banking," Spotify, October 6, 2021, https://open.spotify.com/episode/48fT8xMIy7lDtmRiWHZWKZ.
[17] mint, "UBS-Credit Suisse Merger Could See Up to 30% Job Cut Globally: Report," mint, April 2, 2023, https://www.livemint.com/news/world/ubscredit-suisse-merger-could-see-upto-30-job-cut-globally-report-11680423534465.html.
[18] Ronnie Mitra and Karim Chester, "Why Banks Need to Upgrade Their 'Operating System' | Publicis Sapient," www.publicissapient.com, 2023, https://www.publicissapient.com/insights/why-banks-need-to-upgrade-their-operating-system.

that the adoption of digital banking services has accelerated rapidly in recent years across generations, especially due to COVID-19 shutdowns. The COVID-19 pandemic threw kerosine on the kindling which is the rapid adoption of mobile banking, and there is little sign this will ever reverse.[19]

For the purposes of this paper, whether or not the world has reached "peak digitization" in banking is less relevant in and of itself than the impact it will have on those working in the industry. For example, branch closures are increasing across several markets.[20] While many can be reassigned in the event of branch closures or consolidation, the capacity is not unlimited, and some will unfortunately become redundant and lose their jobs. However, it is likely that automation and digitization alone won't lead to mass layoffs. Take the example of financial advisors. At first glance, financial advisors would seem like some of the first to be replaced by automation. In fact, there has been a notable rise of so-called robo-advisors that manage client funds completely independent of a human. These solutions are growing in popularity and charge significantly lower fees compared to traditional advisors.[21]

While some advisor roles will likely become redundant, the role itself will not be fully replaced – *yet.*[22] Despite impressive growth, automated solutions are still a very small part of the overall advisory market. Additionally, the average AUM for robo-advisors is showing signs of plateauing.[23] This suggests that clients with higher levels of wealth are more than happy to pay higher fees for personalized advice. As of the current moment, the intricacies involved are too extensive for individuals with higher wealth to depend exclusively on automated solutions.

Additionally, there are many reasons to be bullish on the profession. More digitization and new tools mean advisors can take on more clients, while providing the same level of service, all at a lower cost for the individual.[24] Therefore, it is less so that *all* financial advisors will be replaced, and more so that those who fail to adapt to the new digital reality will become uncompetitive. There is a classic trope which remains uncredited to anyone specific that goes: "it's not that [*insert job title here*] will be replaced by technology, rather technology will replace those [*insert job title here*] who fail to learn the technology."

The viewpoint that most professions will not be fully automated seems to be the sentiment held by most experts as well. In a recent interview with CNBC, Harvard Business School professor Dr. Tsedal Neeley discussed why workers should be vigilant, but not overly pessimistic about their job prospects. Dr. Neeley discussed how technology has always created more jobs than it eliminates

[19] Karl Dahlgren, "COVID-19 Pushes Digital Banking Adoption to the Tipping Point," BAI, October 1, 2020, https://www.bai.org/banking-strategies/covid-19-pushes-digital-banking-adoption-to-the-tipping-point/.
[20] Reuters, "HSBC to Close 114 Branches in Britain from April 2023," *Reuters*, November 30, 2022, sec. Finance, https://www.reuters.com/business/finance/hsbc-close-114-branches-britain-april-2023-2022-11-30/.
[21] Jasmin Suknanan, "How to Choose between Using a Robo-Advisor and Using a Traditional Financial Advisor to Manage Your Investments," CNBC, October 19, 2022, https://www.cnbc.com/select/robo-advisor-vs-financial-advisor/.
[22] Sam Taube, "ChatGPT Won't Replace Financial Advisors Yet. Here's Why," NerdWallet, February 10, 2023, https://www.nerdwallet.com/article/investing/chatgpt-wont-replace-financial-advisors-yet.
[23] Statista, "Robo-Advisors - Worldwide | Statista Market Forecast," Statista, August 2023, https://www.statista.com/outlook/fmo/wealth-management/digital-investment/robo-advisors/worldwide#assets-under-management-aum.
[24] Irene Huhulea, "How Technology Is Changing Financial Advice," Investopedia, December 7, 2023, https://www.investopedia.com/how-technology-is-changing-financial-advice-4774011.

in the long run, but the nature of those jobs change.[25] With this as the backdrop, it is important to rethink not only business models, but the skills financial professionals need to succeed.

2.2 Skills to Stay Competitive

The next logical question is: what makes the most sense for financial professionals to learn, in order to stay ahead in this changing landscape? To answer this question and for the following analogies, Excel will be used as a comparative example, since it is a widely adopted industry standard tool at the time of writing.

The first criterion to make a technology worth learning is that it has widespread industry adoption. For example, the reason Excel skills are taught in virtually every finance and business is program because that is what is used in practice. This goes hand in hand with market demand. If there is industry adoption, then there is demand for candidates with that skill. From the perspective of students, they will naturally be drawn to programs that offer to teach these skills too.

The second criterion that makes a technology worth learning is its staying power. When it comes to technology, it often better to think about what is *not* going to change rather than what is likely to change. It takes a significant investment of time to learn any new technology, regardless of its purpose. Even software that is designed to be simple to use often has somewhat of a learning curve. To make the necessary investment of time worthwhile, it must be something that won't be obsolete in a few years already.

Another aspect of staying power in an industry is the community of users and support behind the technology. A large ecosystem of practitioners makes the technology less likely to be replaced since it has so many stakeholders. For example, at the time of writing, when googling "financial analysis with Excel," it returns about 211 million results. That same query with SPSS Statistics, another spreadsheet-based data analysis tool, yields only 72.4 million results (see figure 1). There is nothing wrong with SPSS Statistics, but there is more readily available information about Excel. This inherently raises the switching costs since a larger ecosystem begets a larger ecosystem.

The third criterion to make a technology worth learning is it must be learnable in a reasonable amount of time without having a strictly technical background. If the learning curve for the technology is too high, it won't be widely adopted by users and industry. Excel is a technology that strikes this balance quite well. The average learner will take about three months of dedicated study to become relatively proficient in Excel.[26] To become a true "expert" it takes longer, but three months is certainly enough time to learn the core functionalities. More importantly, it is enough time to learn industry relevant use cases which can be directly applied in a job setting. Additionally, most entry-level finance jobs don't want or expect candidates to be hyperspecialized in a specific aspect of finance or technology. Therefore, it doesn't make sense spending too much time learning a very specific technical skill at the expense of theory. The ideal candidate for most

[25] CNBC, "Harvard Professor on A.I. Job Risks: We Need to Upskill Ad Update Business Models," www.youtube.com, February 24, 2023, https://www.youtube.com/watch?v=45nmTMPp2K8.
[26] Alvin Parker, "How Long Does It Take to Learn Excel in 2023 (Tips & Uses)," prosperityforamerica.com, January 30, 2023, https://www.prosperityforamerica.org/how-long-does-it-take-to-learn-excel/.

finance roles is therefore technically proficient and well-versed in theory with a solid enough base to be flexible and learn very quickly.[27]

Lastly, to make a technology worth learning it should be easily integrable with existing systems. A core part of a technology's scalability and relevance is its flexibility.[28] For financial companies, this is especially relevant due to the highly confidential nature of their businesses' data.[29] Another important facet of integration is the disruption that adoption will cause. Especially for banks that are processing a massive volume of sensitive data in real time, this is very important. Any new technology needs to be customizable enough to be useful and accommodate the institutions needs with minimal risk of disruption when integrated. With Excel, it is essentially a blank slate that can be morphed into whatever the needs of the user are. It also integrates with the entire MS Office ecosystem, accepts most data types as inputs, and can have certain features enabled/disabled depending on security preferences. Since Excel is built to be flexible, it can be adapted to many situations. For example, data is exported from an ERP system as a .csv file; it is then processed in excel as a .xlsx file, and then exported as a .xml file for upload to another system.

Given this, one might wonder why learning another skill besides Excel is worthwhile. Excel is an amazing tool and is not going anywhere, but it has its limitations. Firstly, working with large datasets in Excel can become tedious and slow. Secondly, it is part of the closed Microsoft ecosystem. Third-party packages and plugins are largely incompatible with Excel, unlike with open-source software. Additionally, automating tasks in Excel is difficult. Intricate macros are prone to crashing and the built-in VBA scripting language is very difficult to learn. This makes Excel stellar for small datasets and simple tasks but is not well suited for complex modelling and automation.

None of this is to say finance students should not learn Excel, they absolutely should. The takeaway is that *only* learning Excel is no longer enough to be truly competitive. Consequently, another technology that finance professionals should learn is Python because it meets all the outlined criteria and addresses many of Excel's limitations.

The first criteria Python fuliflls is its widespread industry adoption, especially for data analysis and modeling tasks. Secondly, it has staying power. Survey data from the annual GitHub OctoVerse report shows Python is now the second most popular programming language among developers, showing YoY usage growth of 22.5%.[30] This statistic highlights the mass adoption amongst programmers and data scientists Python has seen and the huge community of users behind it. Thirdly, the learning curve is rather low compared to many other programming languages.[31] Python's simple syntax and high level of abstraction make it the perfect choice for coders without a technical background. Lastly, it integrates well with existing systems. For example, Microsoft has even announced integration between Python and Excel directly, which will allow users to write

[27] Clarity Recruitment, "Finance and Accounting Top Performers Have These 7 Traits," Clarity Recruitment, September 25, 2021, https://findingclarity.ca/blog/finance-and-accounting-top-performers/.
[28] FasterCapital, "Compatibility: The Importance of Fitting Innovations into Existing Systems," FasterCapital, November 2, 2023, https://fastercapital.com/content/Compatibility--The-importance-of-fitting-innovations-into-existing-systems.html.
[29] Andrew Scott and Paul Tierno, "Banking, Data Privacy, and Cybersecurity Regulation," Congressional Research Service, February 24, 2023, https://crsreports.congress.gov/product/pdf/R/R47434/2.
[30] GitHub, "The Top Programming Languages," The State of the Octoverse, 2023, https://octoverse.github.com/2022/top-programming-languages.
[31] Thinkful, "How Hard Is It to Learn Python?," Thinkful, July 30, 2020, https://www.thinkful.com/blog/how-hard-is-it-to-learn-python/.

Python scripts directly in a spreadsheet.[32] Also, since Python is a programming language and not a piece of software, it is built to be highly flexible and has integrations with most data-centric software.

Technically, Python in and of itself is not a technology, rather a general-purpose scripting language.[33] Even though the language was created in the early 1990s, it started exploding in popularity more recently in the early 2010s.[34] Interestingly, the name "Python" comes from the British comedy troupe Monty Python. The creator, Guido van Rossum, got inspiration from the troupe's name because it was short and slightly mysterious.[35] While the scripting language doesn't have many comedic applications, it certainly has many applications relevant to finance.

Today, Python has become the language du jour for data science, data engineering, and general task automation.[36] The reason for this is not necessarily that these use cases are specific to Python, but rather the vast assortment of libraries that are specific to Python.[37] There are libraries for virtually everything in Python, especially data science. The purpose of these libraries is to essentially create layers of abstraction between the user and the operation. In this context, *abstraction* refers to code that is handled in the background, which can be utilized through much simpler operations. Take the example of calculating a covariance matrix of stock returns. Without a library, this could get very complicated. With the popular Pandas library, you simply need to type in pd.cov(*data*) to get the output.[38]

Additionally, survey data from HackerRank shows that Python is the most sought-after programming language for FinTech companies and the third most popular for financial firms overall.[39] At first glance, might come as a surprise that Python wouldn't occupy the top spot for all financial firms. There are a few reasons for this: the principal reason being that banks are very large and complex organizations that are managing many different technologies at once. Also, many of the technology systems used by banks today are legacy and written in other programming languages such as C++, C, and Java. It therefore makes sense that Python would rank highest for digital native FinTech companies but not traditional banks. Since FinTech companies are newer and smaller businesses, they don't have to rely on legacy systems like many traditional banks do.

None of this is to say that Python is as easy to learn such as a UI based piece of software like Excel. To get to an intermediate level, a learner without a technical background will likely need at least

[32] Microsoft Excel, "Announcing Python in Excel: Combining the Power of Python and the Flexibility of Excel.," TECHCOMMUNITY.MICROSOFT.COM, August 22, 2023, https://techcommunity.microsoft.com/t5/excel-blog/announcing-python-in-excel-combining-the-power-of-python-and-the/ba-p/3893439.
[33] Python Software Foundation, "What Is Python? Executive Summary," Python, 2023, https://www.python.org/doc/essays/blurb/.
[34] Python Institute, "About Python," pythoninstitute.org, 2022, https://pythoninstitute.org/about-python#:~:text=Python%20was%20created%20by%20Guido.
[35] Coursera, "What Is Python Used For? A Beginner's Guide," Coursera, September 22, 2021, https://www.coursera.org/articles/what-is-python-used-for-a-beginners-guide-to-using-python.
[36] Elena Kosourova, "What Is Python Used For? 7 Real-Life Python Uses," www.datacamp.com, November 2022, https://www.datacamp.com/blog/what-is-python-used-for.
[37] Anirudh Rao, "Top 10 Python Libraries You Must Know," Edureka (Edureka, December 17, 2018), https://www.edureka.co/blog/python-libraries/.
[38] W3 School, "Pandas DataFrame: Cov() Function," w3resource, August 19, 2022, https://www.w3resource.com/pandas/dataframe/dataframe-cov.php.
[39] Ritika Trikha, "Emerging Languages Overshadowed by Incumbents Java, Python in Coding Interviews," HackerRank Blog, August 2, 2016, https://www.hackerrank.com/blog/emerging-languages-still-overshadowed-by-incumbents-java-python-in-coding-interviews/.

six months of study.[43] Despite this, Python is still by far the most logical choice of technologies to learn for aspiring and existing finance professionals. Python is therefore worth learning mainly because it meets all the criteria outlined in this section and addresses many of the limitations of programs like Excel.

3. How to Learn Python

Performing simple Google searches of topics related to Python or searching for tutorials on YouTube can instantly become overwhelming. There are so many resources out there it can be hard to know where to begin. Given that there is so much information available, students should work backwards. A learner should first identify what their end use cases are, and then learn the tools relevant to those use cases.[44] For most finance professionals, this will be data analysis, forecasting, and modelling.

The first step beyond learning the basic syntax should be getting familiar with the most popular Python libraries for data science. Some of the most prominent data science libraries are NumPy, Pandas, Scikit-learn and Matplotlib.[45] These libraries provide users with out-of-the-box functions that allow for the implementation the complex functionality with only a few lines of code. There is some overlap between these libraries, especially NumPy and Pandas, but they all serve different purposes.

NumPy is the fundamental Python library for numerical computing. This library is especially useful when handling array and matrix operations, which are very common in financial analysis.[46] Pandas, on the other hand, is a library primarily designed for data manipulation, cleaning, and analysis.[47] By far the most useful tool in the Pandas library is the DataFrame. A DataFrame is a two-dimensional data structure made up of columns and rows.[48] It is basically a spreadsheet on which complex mathematical operations can be directly applied. Scikit-learn is a library built on top of Numpy, and another library called SciPy.[49] The mechanics of its building blocks are not relevant, but scikit-learn incorporates features from both of these libraries to make it easy to deploy machine learning and regression models. Finally, Matplotlib is a popular data visualization library which allows users to make graphs and figures.[50] There is a learning curve for Matplotlib, since the syntax is not very intuitive. However, an advantage is the code for a plot often only needs to be set up once and can then be re-used across scripts.

---

[43] avcontentteam, "How Long Does It Really Take to Learn Python?," Analytics Vidhya, April 20, 2023, https://www.analyticsvidhya.com/blog/2023/04/how-long-does-it-really-take-to-learn-python/#:~:text=To%20become%20an%20expert%20in%20Python%2C%20you%20must%20invest%20at.
[44] Geeks for Geeks, "Best Way to Start Learning Python - a Complete Roadmap," GeeksforGeeks, January 30, 2020, https://www.geeksforgeeks.org/best-way-to-start-learning-python-a-complete-roadmap/.
[45] Projectpro, "Top 15 Python Libraries for Data Science and Machine Learning," ProjectPro, October 12, 2023, https://www.projectpro.io/article/top-5-libraries-for-data-science-in-python/196.
[46] NumPy Organization, "NumPy User Guide — NumPy V1.22 Manual," numpy.org, 2022, https://numpy.org/doc/stable/user/index.html#user.
[47] W3 School, "Pandas Introduction," www.w3schools.com, accessed December 17, 2023, https://www.w3schools.com/python/pandas/pandas_intro.asp#:~:text=Pandas%20is%20a%20Python%20library.
[48] Mirko Stojiljković, "The Pandas DataFrame: Make Working with Data Delightful – Real Python," realpython.com, 2021, https://realpython.com/pandas-dataframe/#:~:text=The%20Pandas%20DataFrame%20is%20a.
[49] Ashish Choudhary, "15 Most Important Features of Scikit-Learn!," Analytics Vidhya, July 15, 2021, https://www.analyticsvidhya.com/blog/2021/07/15-most-important-features-of-scikit-learn/#:~:text=Scikit%2DLearn%2C%20also%20known%20as.
[50] Matplotlib, "Tutorials — Matplotlib 3.5.0 Documentation," matplotlib.org, 2012, https://matplotlib.org/stable/tutorials/index.

All these libraries are very large and have too many functions to ever memorize fully. Luckily, since they are so popular, finding information about any specific method is only one Google search away. Thanks to these libraries, students can think at a much higher level and focus more of their time on learning the basic syntax of Python. Even with only a basic grasp of the syntax, these libraries can be effectively implemented.

Once a student has learned the basics and familiarized themselves with these libraries, the next step is to jump in and start building projects.[51] A key aspect of a programming is that there is little ambiguity, the script will either work as intended or it produces an error. There are theoretical aspects to Python, but more so related to various code design philosophies than actual implementation.[52] Since there isn't a large theoretical aspect to the actual implementation of Python code, learning by doing is often the best approach.

Another reason building projects is the best learning approach for finance students is they dynamically encounter problems. In programming, knowing how to search for an answer is almost as important as understanding the concept itself. According to survey data, 63% of programmers spend more than 30 minutes every day searching for solutions.[53] It is more efficient for students to get familiar with documentation and independent search than trying to memorize every function they could possibly need.

The use of case studies thereby creates a self-perpetuating feedback loop of learning where the student encounters a problem, then finds a solution, and so on. A programming project is nothing more than a large collection of small problems that need to be solved one at a time. After solving the problem once, when a similar situation arises in the future, the student will have the mental model to approach the problem successfully. For example, there are only so many ways to set up a regression model. The data may change, but the fundamentals of running a regression model with a library such as Scikit-learn won't vary widely across projects.

To put it all together, Python is a large and multifaceted language with almost limitless use cases. Therefore, students should start with their specific use cases and then work backwards from there to identify what is worth learning. Once the use cases have been established, students should focus on understanding the libraries that will help them achieve these goals. After learning the fundamentals, students can begin building industry relevant case studies. With these steps, students can use the time they invest in learning Python as efficiently as possible. This way, students are building tangible skills that can be applied directly in an industry setting.

3.1 Creating Effective Case Studies

To create an effective Python case study, it must fulfill certain criteria. The first criterion is that the case study has a clear set of results: for example, a case study where students need to give a buy or sell recommendation for a stock. Another important criterion is that the case has concrete learning outcomes for the students. A case study is different from a course in the sense that the

[51] Matt Crabtree and Adel Nehme, "How to Learn Python from Scratch in 2023: An Expert Guide," Datacamp.com, July 1, 2023, https://www.datacamp.com/blog/how-to-learn-python-expert-guide.
[52] Vahndi Minah, "Data Science and Design Thinking Belong Together," frog, part of Capgemini Invent, accessed December 17, 2023, https://www.frog.co/designmind/data-science-and-design-thinking-belong-together.
[53] Statista, "Time Developers Spend Searching for Solutions Worldwide 2023," Statista, October 26, 2023, https://www.statista.com/statistics/1401435/daily-time-spent-searching-solutions-developers-globally/.

focus is much narrower. The goal should be to complement the material being learned in the accompanying course or degree program with a deeper dive into specific areas.

Additionally, the case study should have clear performance criteria. In a case study setting, the performance criteria should be tied to the intended learning outcomes. For example, evaluating both the quality of the student's code as well as the analysis. If the code is very sophisticated, but the quality of analysis is not, then the student hasn't achieved the learning outcome. Lastly, the case study should have real-world applicability in the job market. Case studies are a great opportunity to put theory into practice and tie together what is learned in the classroom with what is expected in a job setting.[54]

In a setting where the case studies are done in tandem with a specific course, the purpose should be to apply the knowledge of the course in a Python environment. If it is for a workshop, then the case study can be tailored to the topic at hand. Regardless, the learning outcomes for Python case studies should be related to a concept's application rather than learning a new concept altogether. For example, students who are in a statistics course could learn how to apply significance tests in Python, rather than the theory behind significance testing. The case study should therefore focus more heavily on application than introducing a new concept. Given these criteria, the subsequent section gives an overview of three possible Python case studies for finance students.

4. Three Python Case Studies

Before exploring these case studies, a few assumptions are considered. The first assumption is that the participating students have a basic understanding of Python' syntax. Secondly, considering this proficiency level, the coding component of each case study is structured to take approximately 10-15 hours. The case studies are all structured to have two components: the code and a report. The first part is the code itself as well as any data visualizations. The code should be heavily commented in the workbook, clearly explaining the purpose each block of code serves. The second part of the report is a written discussion of the findings. The written portion should include both the student's approach to writing the code as well as a review of the analysis's findings. Finally, the case studies all tie into topics discussed in existing modules, so the topics should not be foreign to Master of Science in Banking and Finance students at HSLU.

4.1 Binary Classification: Predicting Credit Default of Retail Banking Customers

Problem Statement:

Interest income is one of the main drivers of bank profits.[55] Inherent to almost every financial institution is a profound concern for estimating the likelihood of default when issuing a loan. There is growing evidence that machine learning and binary classification algorithms can be used to accurately predict default probability.[56]

[54] Monash University, "Writing a Case Study," Monash.edu, accessed December 17, 2023, https://www.monash.edu/student-academic-success/excel-at-writing/how-to-write/case-study.
[55] MX, "How the 4 Biggest U.S. Banks Generate Income and Revenue | MX," mx.com, July 14, 2020, https://www.mx.com/blog/top-us-retail-banks-income-revenue/.
[56] Yashna Sayjadah et al., "Credit Card Default Prediction Using Machine Learning Techniques," *2018 Fourth International Conference on Advances in Computing, Communication & Automation (ICACCA)*, October 2018, https://doi.org/10.1109/icaccaf.2018.8776802.

Dataset: Credit Risk Dataset (kaggle.com)[57]

Prompt:

Download and explore this Kaggle dataset from a Portuguese bank. The dataset contains various information about borrowers as well as whether they are currently in default or not. Start with exploratory data analysis. Once the data has been cleaned, set up a logistic regression model using credit defaults as the target variable. Pay attention to the individual variables and think about whether they need to first be transformed. Are there any redundant variables which can be excluded? After setting up your initial model, apply various penalties and explore how this impacts your model's ability to generalize given your chosen evaluation criteria. Throughout the assignment, ensure to add several comments in your code using the # sign so it is comprehensible to someone who has not seen your code before. Then create visualizations of your results to include in your discussion. Finally, when discussing your findings, give concrete and actionable advice to the bank in question that loan officers can use when making loan decisions.

Questions to Consider:

- *What is your opinion of the dataset? Is it balanced? Do the independent variables make sense? Are any of them redundant?*
- *How effective was your regression model at predicting default? How did penalties impact results? What metrics did you use to evaluate your model?*
- *Based on your model, what is the most significant predictor of default?*

Learning Outcomes:

Students who complete this case study will gain industry relevant knowledge for predicting credit default of retail banking customers. Students will better understand the relationship between commonly collected customer data points and their impact on credit risk, similar to those banks use to evaluate creditworthiness in practice. Additionally, students will better understand model specification as well as how to identify redundant variables. By redundant variables, this refers to variables that capture several pieces of information in one metric, instead of using several different metrics which essentially capture the same information in aggregate. Lastly, the semi-structured nature of the dataset gives students the opportunity to perform exploratory data analysis. This is important since the quality of the inputs in a model have an enormous impact on its overall ability to generalize and minimize over/underfitting.

Sample Results Summary:

A correctly specified model will likely identify the loan-to-income ratio as the most significant predictor of credit default. In this sample analysis, the base model without penalties identifies the loan recipient's status as a renter to be the most powerful predictor of default (see figure 2).

[57] https://www.kaggle.com/datasets/laotse/credit-risk-dataset

However, once various penalty terms are applied, the loan-to-income ratio (LI) is selected as the most important independent variable by a significant margin (see figure 3).

The models suggest that once a loan exceeds 30% of annual income, the default probability increases significantly. However, whether the borrower is a renter, as well as the loan grade are still identified as somewhat relevant factors. This intuitively makes sense as there is a plethora of information captured in these variables. If someone is renting, they are likely of lesser means than someone who is a homeowner. This increases the likelihood of future repayment issues.

Additionally, a column including the loan grade assigned by the bank is included in the dataset. In the description of the dataset itself, there aren't specific details that outline the bank's loan grading system, but the model suggests it seems to be effective. The lower the grade, the higher the chance of default. While it is not as strong of a relationship as the LI, there is certainly a correlation between a lower loan grade and default. This suggests the bank's internal grading system is effective at identifying lower quality applicants but needs some improvement.

Another interesting observation from this analysis is that the default probability for those in 10-20% LI bin is lower overall, but the share of defaults is comparable with the 30-40% LI bin (see figure 4). This shows that low-quality borrowers have trouble repaying a loan even if it is a lower share of their income, and higher quality borrowers face difficulty making payments when the 30% LI threshold is crossed. The finding suggests borrowers of all quality levels have difficulty making repayments once the 30% LI threshold is crossed. A possible interpretation of this is the negation of the quality of the borrower as a highly relevant factor for larger loans. However, the findings also shows that high quality borrowers are much less likely to default on smaller loans as measured by LI, if the LI remains below 30%.

Another interesting finding is that the base model identifies a prior default as the third most significant default predictor whereas the penalized models largely disregard prior defaults. This suggests that loan applicants who have defaulted in the past are not inherently bad loan candidates, given that other factors suggest a low risk of default. However, a major caveat of all these findings is that the data is from a Portuguese bank and may not generalize to other markets.

Finally, there are several pieces of actionable advice that can be gleamed from this analysis. Firstly, the loan grading system that is currently employed seems to work well, but more weight should be added to the LI ratio. Secondly, a high-quality borrower whose LI is approaching 30% is likely going to face issues with repayment regardless of other quality indicators. Loan officers should therefore focus mainly on ensuring the LI remains below 30% when making lending decisions, regardless of borrower quality.

4.2 Monte Carlo Simulation of Portfolio Returns

Problem Statement:

Accurately predicting a portfolio's return is difficult if not impossible. Regardless of the efficient market hypothesis's validity, randomness plays a significant role in security returns, especially in

the short to medium term.[58] Instead of trying to correct for randomness, it is often better to utilize a model which incorporates it. This partially explains the popularity of Monte Carlo simulations for portfolio modelling. This type of simulation is a technique that leverages random sampling and probability to model complex outcomes.[59] In the context of finance, this means estimating volatility and returns by averaging out many possible paths that incorporate randomness. A Monte Carlo simulation can be done on a set portfolio, or with random portfolio weights for each path. If the simulation is done with random weights, an analyst can compare many different possible scenarios. Regardless of the initial approach, when the simulation's paths are created, valuable information can be extracted from the many runs.

Prompt:

Familiarize yourself with Monte Carlo simulations. Try to understand the underlying mathematical operations and assumptions. Next, think about which Python libraries you will need to run such a simulation. Once you understand the mechanics, create a portfolio of ten publicly traded companies to perform the analysis on. Using this portfolio, gather historical data and run a Monte Carlo simulation with at least 500 paths. Assume no short selling and a time horizon of one year (252 trading days). Ensure the portfolio weights for each run are randomly selected. To collect the historical data, try using the YahooFinance API. Plot and discuss your results and ensure your code is heavily commented and each block of code is well explained. Finally, make a recommendation for the portfolio you created. Would you recommend investing in this basket of stocks? If so, what weights does the simulation suggest? What criteria did you use to extract the ideal portfolio?

Questions to Consider:

- *Explain your approach to portfolio construction. How did you choose your allocations?*
- *Discuss the advantages and disadvantages of Monte Carlo simulations. Is there a better way to achieve the same result? Are the assumptions realistic?*
- *What resources did you use to build your model? What assumptions did you use when building your model?*
- *Discuss your model's code. What steps did you take to pre-process the data? How did you set up the simulations? How did you set up the simulation loop?*
- *Discuss your results, what is the mean return of your portfolio? Are you surprised by these results?*
- *What portfolio had the maximum Sharpe Ratio? Does the simulated Sharpe differ significantly from the historical version? What about its return?*

[58] Paul Kosakowski, "Financial Markets: Random, Cyclical or Both?," Investopedia, August 22, 2022, https://www.investopedia.com/articles/financial-theory/09/markets-cyclical-vs-random.asp.
[59] Will Kenton, "Monte Carlo Simulation: History, How It Works, and 4 Key Steps," Investopedia, November 2, 2023, https://www.investopedia.com/terms/m/montecarlosimulation.asp#:~:text=%25%2025%25%200%25-.

Learning Outcomes:

Students who participate in this case study will gain valuable knowledge of data analysis coding concepts and the application of financial modelling techniques. Monte Carlo simulations are widely used in industry and are an important tool for simulating portfolio returns and volatility. Students who complete this case study will also become familiar with the YahooFinance API, which is simple to use and has widespread adoption among practitioners. Creating a Monte Carlo simulation in Python enhances students' proficiency in connecting theoretical concepts with practical applications, bridging the gap between academic knowledge and industry demands. Additionally, students will gain valuable knowledge of setting up for loops, data visualization, data preparation, and the application of statistical mathematics in a Python environment.

Sample Results Summary:

The portfolio used in this case consisted of ten technology companies: Amazon, Apple, Microsoft, Google, Taiwan Semiconductor, Tesla, Nvidia, Intel, Adobe, and Netflix. An annual risk-free rate of 4% was assumed for calculating the Sharpe ratios, and there were 1000 paths created over a trading year's timeframe of 252 days (see figure 5). The goal of the analysis was to construct a portfolio of high-risk high-reward stocks. Historical data was collected using the YahooFinance API going back ten years.

The results are somewhat unsurprising, with the simulated portfolios averaging a mean annual return of around 29% and annual volatility of 26%. This is not a surprising result because, at the time conducting the analysis, the market is coming off the longest technology bull run in history. This, in turn biases the data to the upside.[60] Additionally, the mean Sharpe ratio of the simulated portfolio was 0.96, which is 33% higher than the technology heavy NASDAQ100 index's ten-year historical Sharpe ratio of 0.72.[61]

Another interesting result of the Monte Carlo simulation was the divergence from the maximum Sharpe ratio's portfolio statistics in the simulation setting versus historical calculations. Based on the simulated results, the maximum Sharpe ratio's one year return is 191% and has a standard deviation of 25%. This results in a very high Sharpe ratio of 7.54. If only investing were this easy! The largest two holdings for this portfolio are Google and Amazon (see figure 6). The findings are not necessarily surprising but should be approached with a prudent level of skepticism. Clearly, these returns are very unrealistic, but the weights identified by the simulation can still be used to construct a realistic portfolio to estimate its return, standard deviation, and Sharpe ratio.

Using historical returns and the weights from the max Sharpe portfolio, the one-year return is estimated to be 23% with a standard deviation of 25%. These results indicate an adjusted Sharpe ratio of 0.74, which is about the same as the historical Sharpe ratio for the NASDAQ100. Lastly, while the findings of the historical analysis are logical, they represent the predictions of a single snapshot in time. This ties into a common criticism of Monte Carlo simulations, namely that they are not reliably accurate at predicting returns. Despite this, they continue to be used in industry.

[60] Alex Wilhelm, "Was Tech's 'Bull Run' Simply a Temporary Surge?," TechCrunch, September 27, 2023, https://techcrunch.com/2023/09/27/was-techs-bull-run-simply-a-temporary-surge/.
[61] Morningstar Research, "NASDAQ Composite PR USD (@CCO) Risk | Morningstar," Morningstar, Inc., December 17, 2023, https://www.morningstar.com/indexes/xnas/@cco/risk.

Where Monte Carlo simulations shine is in modelling volatility, and this is often what practitioners use Monte Carlo simulations for.[62]

Finally, based on this analysis, the max Sharpe portfolio should not receive a buy recommendation. The risk reward tradeoff does not warrant investing in this portfolio, especially when investors can invest in passive index fund which already contains all these securities. Investors should therefore consider investing in one or two of the portfolio's stocks and pair this with a passive index fund which would likely achieve similar returns with a more efficient risk-return relationship.

4.3 Cross-Sectional Analysis of Stock Market Returns

Problem Statement:

The stock market's returns and broader economic indicators have a complex relationship. Research has shown that certain metrics like the unemployment rate and the level of retail sales have an impact on stock market returns.[63] With advancements in computing, machine learning models and advanced analytics operations can now be done by anyone with a laptop. Not only that, but research has also shown stock market returns can be somewhat accurately predicted using machine learning models in a development environment.[64] Given these advancements and prior research, it is now easier than ever to analyze the relationships between various economic indicators and returns in financial markets.

Prompt:

Explore various economic indicators that have demonstrated historical correlations with stock market performance. After settling on which economic indicators to employ, choose a relevant stock market index to use as your target variable. Ensure the indicators are relevant to the index. For example, use a German market index if you are using German economic indicators. Next, source your data for the stock index. The YahooFinance API is a good place to start as it has historical data for most major indices. Then collect the data for the various economic indicators. Ensure the intervals for each indicator and the target variable match. If you are using US data, the FRED API is simple to use and has data for most relevant economic indicators. After collecting and processing your data, choose an appropriate machine learning algorithm to predict the market's historical returns. *Hint:* find algorithms that generalize well on small datasets. Finally, create a plot of the predicted versus actual returns and discuss your model's ability to generalize.

Questions to Consider:

- *Discuss the economic indicators and market index used. Why did you choose the indicators/index? Where did you source your data from?*

---

[62] David Blanchett, "A Key Criticism of Monte Carlo Is Unfounded," Advisor Perspectives, August 31, 2023, https://www.advisorperspectives.com/articles/2023/08/31/income-criticism-monte-carlo-david-blanchett#:~:text=Indeed%2C%20approximately%2080%25%20of%20financial,follow%20a%20perfect%20bell%20curve.

[63] Aaron Levitt, "Economic Indicators That Affect the U.S. Stock Market," Investopedia, January 23, 2019, https://www.investopedia.com/articles/investing/031413/economic-indicatiors-affect-us-stock-market.asp.

[64] Mehar Vijh et al., "Stock Closing Price Prediction Using Machine Learning Techniques," *Procedia Computer Science*167, no. 167 (2020): 599–606, https://doi.org/10.1016/j.procs.2020.03.326.

- *What algorithm did you use? What factors did you consider when choosing it? Were there any limitations to your implementation?*

- *Which factors did your model identify as having the largest impact on stock market returns? Do you agree with the findings?*

- *How did you correct for model misspecification and overfitting?*

- *What metrics did you use to evaluate your model? How did you decide which metrics to use?*

- *How do you think your model would perform in a real-world setting? Is your model realistic?*

Learning Outcomes:

Students who participate in this case study will gain valuable knowledge of cross-sectional analysis, working with financial APIs and implementing machine-learning models. Since it is relatively straightforward to setup and implement a machine-learning model, understanding what model to use in each context is just as important as the implementation. The process of model selection will enable students to learn more about machine-learning algorithms in general. Through this research, students will gain better understanding of the relationship between input data and model selection.

Additionally, because these models are so powerful, understanding which inputs to use is incredibly important to build a model that generalizes well. Oftentimes machine-learning models will falsely identify relationships in a development environment, which then don't hold in production. Therefore, critically thinking about the selection of independent variables is an important learning outcome. Also, incorrectly calibrated machine-learning models are highly prone to over- or underfitting. Unfortunately, it is very easy to set up a model that might appear to work well but is failing to generalize.[65] Part of creating an effective machine-learning model is learning how to tweak the parameters beyond the base model to correct for these issues. This is a concept from statistics that students will already be familiar with but likely not have explored in depth with machine-learning models in Python environment.

Sample Results Summary:

The following independent variables were used in this sample analysis: the ten-year treasury rate, the producer price index (PPI), the consumer price index (CPI), the unemployment rate, the change in retail sales, the level of consumer confidence, and the Case Shiller US real estate index. All these indicators are in a monthly interval and were retrieved from the FRED API. The target variable is monthly returns of the S&P 500. The analysis employs these data inputs covering the period from February 1985 to October 2023. Since the dataset is relatively small, a random forest model was employed to predict the monthly returns of the S&P 500.

[65] SuperAnnotate, "Overfitting and Underfitting in Machine Learning | SuperAnnotate," www.superannotate.com, October 17, 2022, https://www.superannotate.com/blog/overfitting-and-underfitting-in-machine-learning#what-is-overfitting.

The data was first split into a test and training set with 70% of the data used for training and the remaining 30% for testing. A cross validation (CV) object was then employed to reduce the risk of overfitting. The CV object employed five splits. Since the dataset is rather small, an n_estimators value of 100 is passed in to further reduce the risk of overfitting. Finally, the max_features hyperparameter was set to the square root of the number of features. While increasing the number of features often leads to better performance, it also increases the risk of overfitting.[66] By taking the square root, the model can strike a balance between feature diversity and avoiding overfitting.

The mean absolute error (MAE) and mean squared error (MSE) were used as evaluation metrics. These metrics are useful because the MAE is less sensitive to outliers and provides a more intuitive understanding of the average prediction error, while the MSE gives more weight to larger errors. By using both metrics, one can gain a better understanding of the model's overall performance.

Despite the relatively small sample size of 380 observations, the model performed reasonably well. The average MAE was 3.25% with a standard deviation among the CV models of 0.1%. This suggests the model has some predictive ability but is not overfit. The model also returns an average MSE of 0.2% and a standard deviation among CV models of 0.016%. It is not unsurprising to see such a divergence between the MAE and MSE values because, since the dataset is small, it makes the model more sensitive to outsized market movements. This is evident by looking at a plot of the predicted versus actual values (see figure 7). The model seems to have trouble identifying large swings but performs well under normal conditions. This suggests the model can be effective during calm market conditions but not provide valuable insights when volatility is increasing in the broader market.

4.4 Relevant Modules Discussion

The case studies outlined above were crafted with the HSLU Msc. Banking and Finance modules in mind. However, the case studies can also be done independently in the form of a standalone workshop or extracurricular assignment. There are three principal modules where these case studies can be incorporated, namely Research Methods, Risk Management, and Equity Investments modules. In the current Research Methods modules, the existing assignment could also likely be directly transferred to a Python environment. To build on these assignments, however, the cross-sectional returns analysis case study fits into the module's curriculum as it currently stands.

The Risk Management module also has much potential for Python integration into its assignments. As a field, risk management is incredibly exciting because it is currently being transformed by machine learning and data analysis.[67] Market Risk, Credit Risk, and Risk Management in Banking all have potential for Python integration in industry relevant ways. As it stands, the case most relevant to this module is naturally the binary classification exercise, but there are many other possibilities for further case study development. Additionally, in its current state, many of the assignments for the Equity Investments module are done in Excel and can also be directly

[66] Tavish Srivastava, "Tuning the Parameters of Your Random Forest Model," Analytics Vidhya, June 9, 2015, https://www.analyticsvidhya.com/blog/2015/06/tuning-random-forest-model/#:~:text=sqrt%20%3A%20This%20option%20will%20take.
[67] Deloitte Advisory, "Why Artificial Intelligence Is a Game Changer for Risk Management," 2016, https://www2.deloitte.com/content/dam/Deloitte/us/Documents/audit/us-ai-risk-powers-performance.pdf.

transferred over to Python. However, the Monte Carlo simulation exercise could be integrated quite well with the current curriculum of the Equity Investments module.

The sample case studies were designed to be generally applicable across the Msc. Banking and Finance program. These case studies can be treated as out-of-the box assignments or used as a blueprint that can be implemented for bespoke case studies. However, it goes without saying that a course is at the sole discretion of the individual teaching it, and it is ultimately a decision individual faculty need to make in terms of if and how they integrate the case studies in their courses. The purpose of these case studies is to highlight the potential for synergies between learning Python and the Msc. Banking and Finance program's existing curriculum.

In summary, the case studies suggested here are meant to prepare students for using Python in practical applications. The case studies aim to incorporate concepts learned in the Msc. Banking and Finance modules into the Python environment. By participating in these case studies, students will gain valuable knowledge of the key libraries used in the field of finance that have direct industry application, while also engaging with the learning outcomes associated with each case study. The main assumption underpinning these case studies is that students have a basic grasp of the Python syntax. The coding aspect is designed to be challenging, but doable in reasonable amount of time for a novice coder. Finally, the analysis section in each case study is designed to encourage students to engage critically with their code and output, rather than merely setting up a model and reciting the results.

5. Conclusion

Overall, this paper has examined several aspects of technology and how it pertains to aspiring finance professionals. While many feel anxiety about their future job prospects given the rapid technological advancements being made in the finance industry, experts tend to agree most jobs won't be fully replaced.[68] Finance students should therefore feel confident in following their passion, regardless of the direction it takes them in finance. However, they can't risk burying their heads in the sand and pretend nothing will change, or their specific role of choice will not be affected by automation – it will. Students therefore need to be ever cognizant of the changes taking places and adjust accordingly.

Students should focus on learning industry relevant technical skills that have a positive return on investment as defined by widespread industry adoption, staying power, integrability with existing systems, and a manageable learning curve. For finance students, this technical skill is undeniably Python.[69] Since Python is such a large language, students should focus first on the end-application and learn the facets of Python apply to that subset of use cases. To achieve this, students should focus on building projects which apply financial concepts in a Python setting.

Finally, while learning a programming language is exciting, it is not easy. The three potential case studies outlined in this paper are not a replacement for diligent study of Python's core concepts. Despite Python being comparatively easy to learn, the key word is still *comparatively*.

[68] Lauren Leffer, "'AI Anxiety' Is on the Rise--Here's How to Manage It," Scientific American, October 7, 2023, https://www.scientificamerican.com/article/ai-anxiety-is-on-the-rise-heres-how-to-manage-it/#:~:text=Alvord%20says%20her%20clients%20of.

[69] Adrian Bosacki, "How Is Python Used in Finance? — Python Applications in Finance," www.softkraft.co, September 15, 2021, https://www.softkraft.co/how-is-python-used-in-finance/.

Programming is incredibly rewarding and can be a very lucrative skill, but anyone claiming it can be learned with one, two, or even three solid case studies is lying. If a student wants to learn how to code, it will take dedication and practice, there is no getting around that. With that said, it is absolutely worth the investment of time to learn Python for aspiring finance practitioners.

Despite the challenges students face in this ever-changing landscape, they should not be nervous about technology but be excited by it. With that said, not being proactive simply isn't an option anymore. Change brings opportunity, but it is on the individual to take advantage of them. Instead of being intimidated and feeling defeated, we should learn to embrace change and ambiguity. After all, we don't have any other choice, and faculty can support students' confidence in terms of learning more about and how to use technology within finance.

6. Appendix

Figure 1: Excel vs. SPSS Statistics Google Query Result

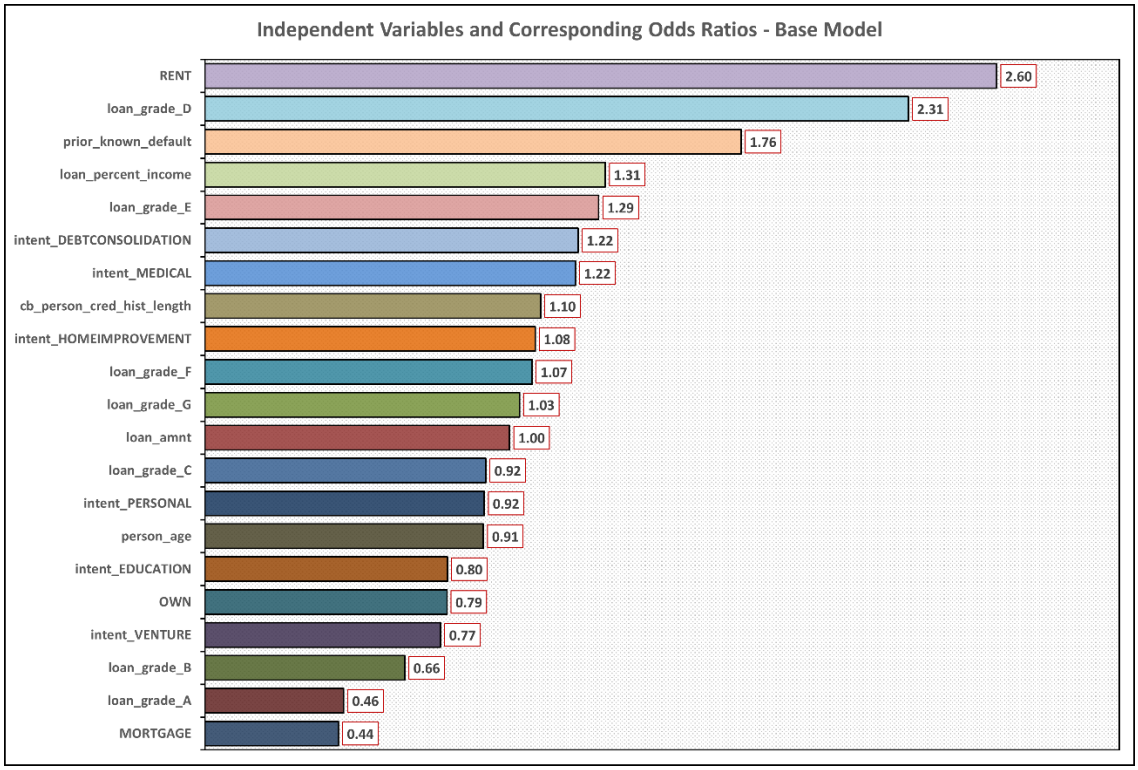

Figure 2: Base Model Logistic Regression Odds Ratios

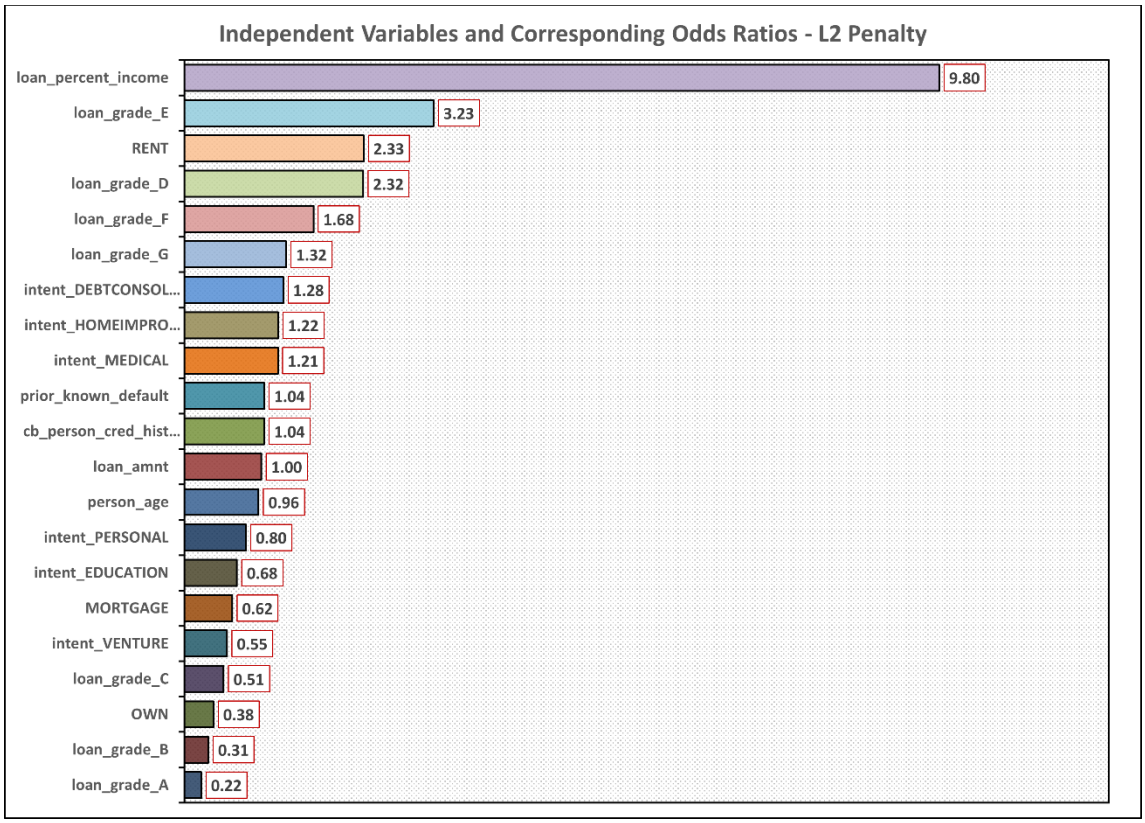

Figure 3: Logistic Regression Model w/ L2 Penalty Odds Ratio

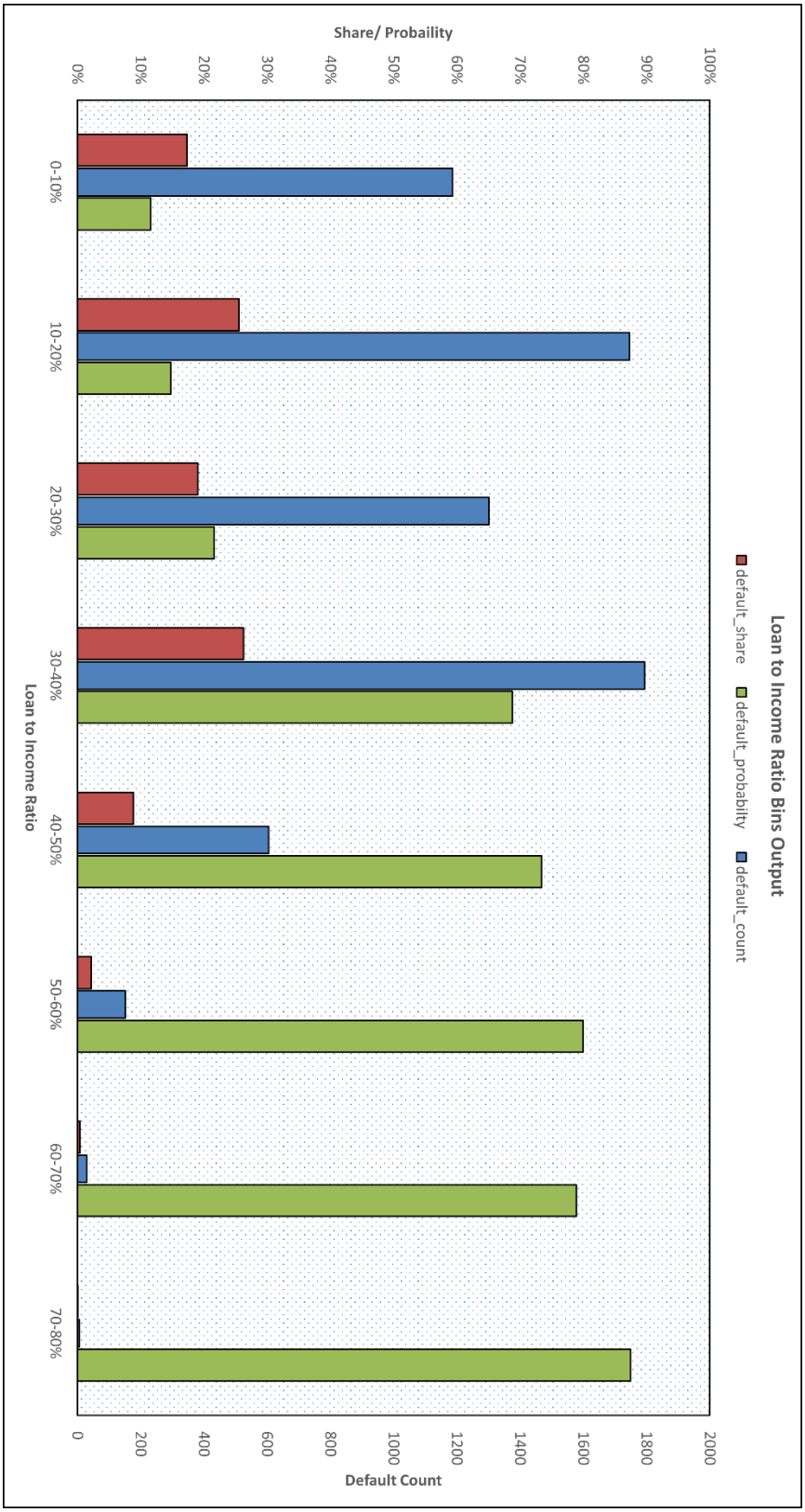

Figure 4: Loan to Income Ratio Bin Graph

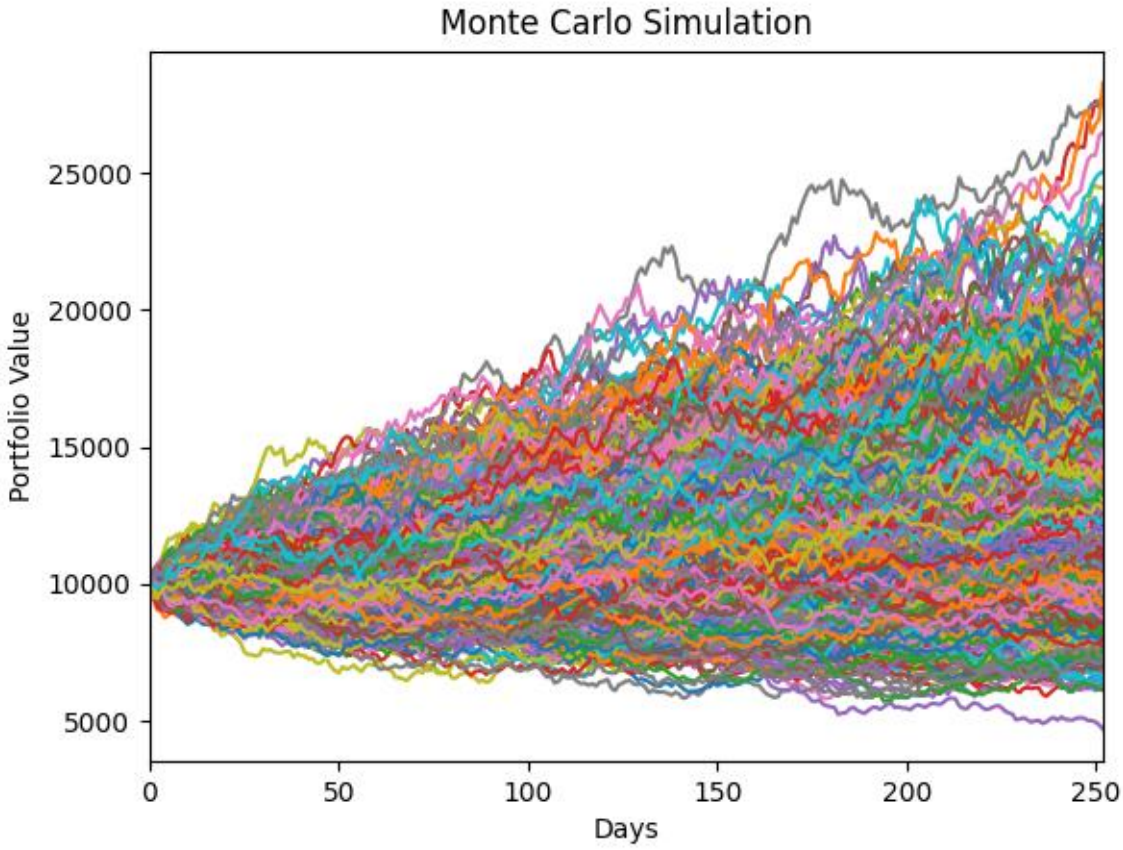

Figure 5: Monte Carlo Simulation Output

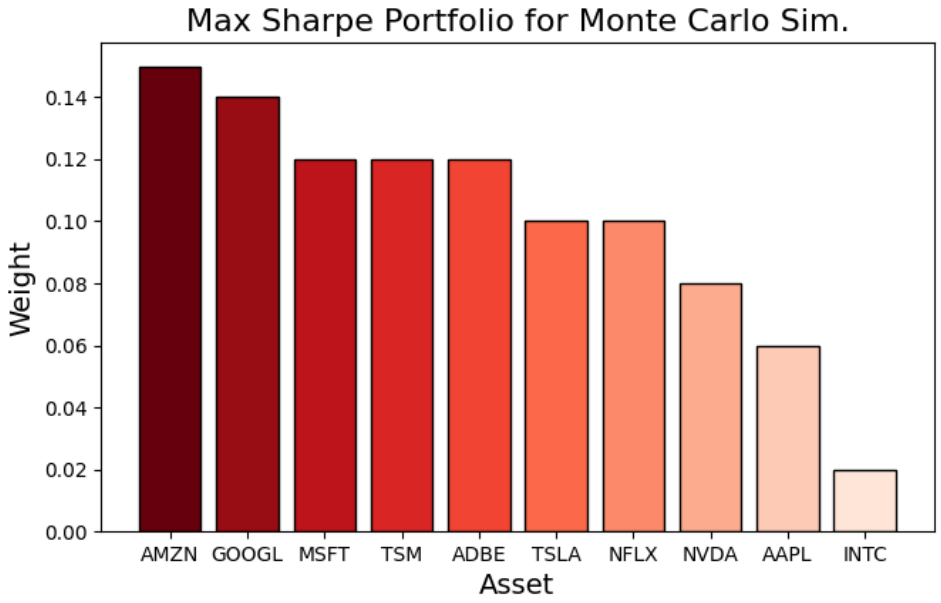

Figure 6: Maximum Sharpe Ratio Portfolio Weights

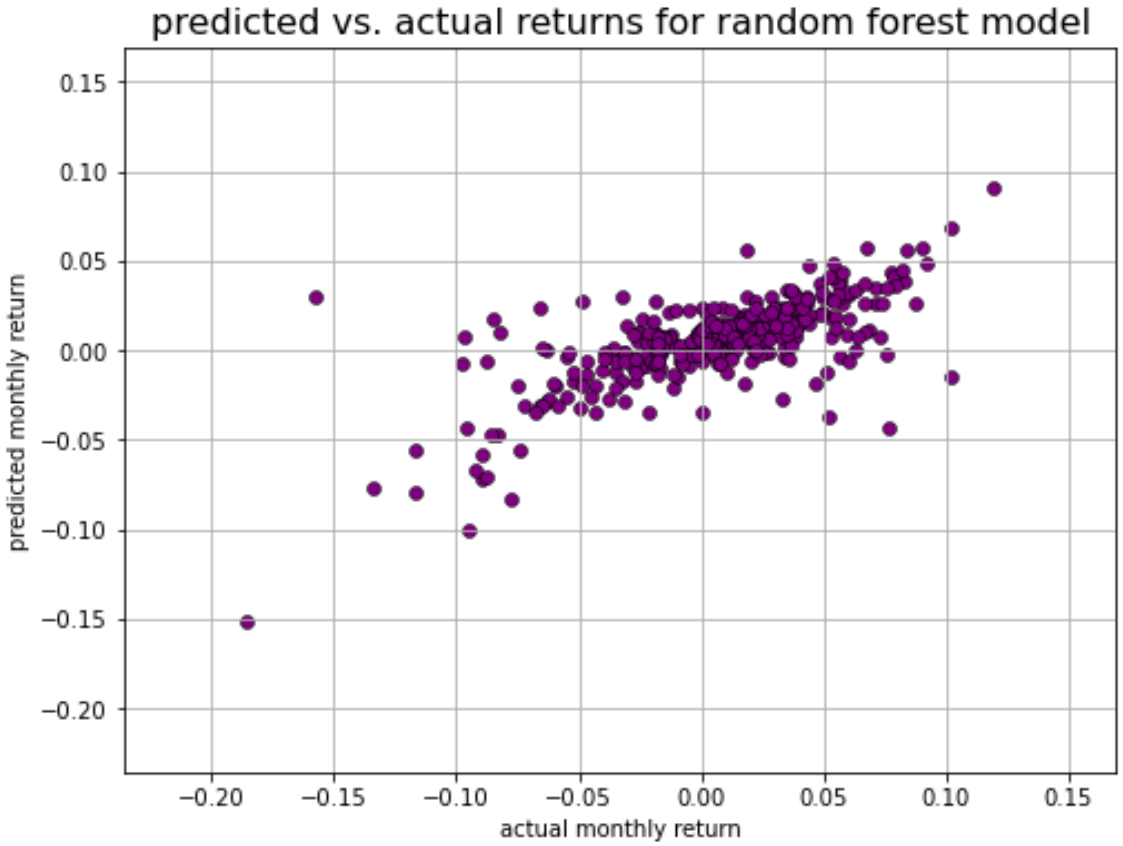

Figure 7: Predicted vs. Actual Monthly Return Plot

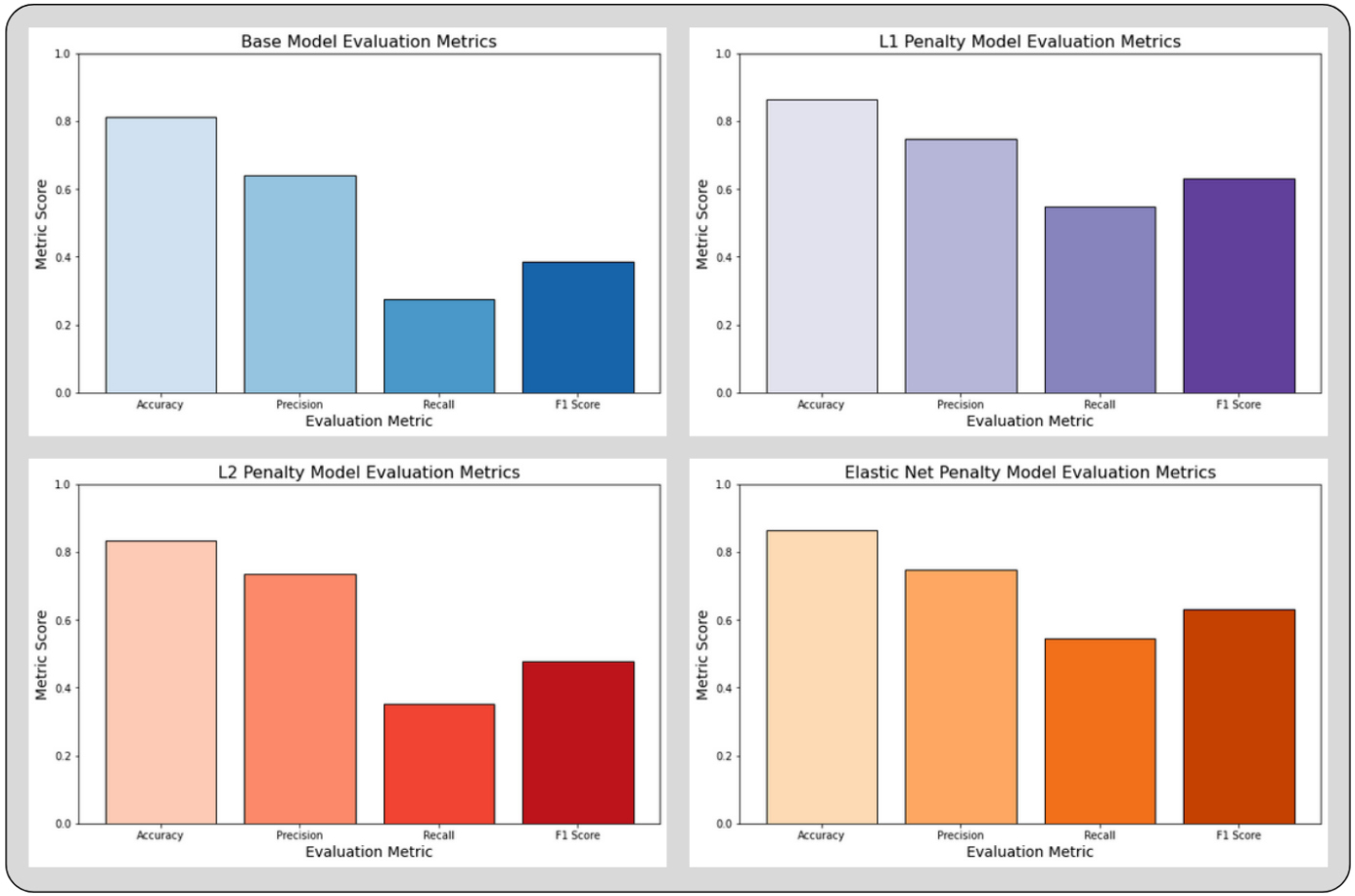

Figure 8: Logistic Regression Model Accuracy Metric Scores